

The Capacity of Private Information Retrieval

Hua Sun and Syed A. Jafar

Abstract

In the private information retrieval (PIR) problem a user wishes to retrieve, as efficiently as possible, one out of K messages from N non-communicating databases (each holds all K messages) while revealing nothing about the identity of the desired message index to any individual database. The information theoretic capacity of PIR is the maximum number of bits of desired information that can be privately retrieved per bit of downloaded information. For K messages and N databases, we show that the PIR capacity is $(1 + 1/N + 1/N^2 + \dots + 1/N^{K-1})^{-1}$. A remarkable feature of the capacity achieving scheme is that if it is projected onto any subset of messages by eliminating the remaining messages, it also achieves the PIR capacity for that subset of messages.

1 Introduction

Introduced in 1995 by Chor, Kushilevitz, Goldreich and Sudan [1, 2], the private information retrieval (PIR) problem seeks the most efficient way for a user to retrieve a desired message from a set of distributed databases, each of which stores all the messages, without revealing any information about which message is being retrieved to any individual database. The user can hide his interests trivially by requesting all the information, but that could be very inefficient (expensive). The goal of the PIR problem is to find the most efficient solution. Here is the problem description. We have N non-communicating databases, each stores the full set of K independent messages W_1, \dots, W_K . A user wants one of the messages, say $W_i, i \in \{1, 2, \dots, K\}$, but requires each database to learn absolutely nothing (in the information theoretic sense)¹ about the retrieved message index, i . To do so, the user generates N queries Q_1, \dots, Q_N and sends $Q_n, n \in \{1, 2, \dots, N\}$ to the n -th database. After receiving query Q_n , the n -th database returns an answering string A_n to the user. The user must be able to obtain the desired message W_i from all the answers A_1, \dots, A_N . To be private, each query Q_n and each answer A_n must be independent of the desired message index, i .

For example, suppose we have $N = 2$ databases and K messages. To retrieve W_i privately, the user first generates a random length- K vector $[h_1, h_2, \dots, h_K]$, where each element is independent and identically distributed uniformly over \mathbb{F}_2 , i.e., equally likely to be 0 or 1. Then the user sends $Q_1 = [h_1, h_2, \dots, h_i, \dots, h_K]$ to the first database and $Q_2 = [h_1, h_2, \dots, h_{i-1}, (h_i + 1), h_{i+1}, \dots, h_K]$

Hua Sun (email: huas2@uci.edu) and Syed A. Jafar (email: syed@uci.edu) are with the Center of Pervasive Communications and Computing (CPCC) in the Department of Electrical Engineering and Computer Science (EECS) at the University of California Irvine.

¹There is another line of research, where privacy needs to be satisfied only for computationally bounded databases [3, 4, 5].

to the second database. Each database uses the query vector as the combining coefficients and produces the corresponding linear combination of message bits as the answer to the query.

$$A_1 = \sum_{k=1}^K h_k W_k \quad (1)$$

$$A_2 = \sum_{k=1}^K h_k W_k + W_i \quad (2)$$

The user obtains W_i by subtracting A_1 from A_2 . Privacy is guaranteed because each query is independent of the desired message index i . This is because regardless of the desired message index i , each of the query vectors Q_1, Q_2 is individually comprised of elements that are i.i.d. uniform over \mathbb{F}_2 . Thus, each database learns nothing about which message is requested.

The PIR problem was initially studied in the setting where each message is one bit long [1, 2, 6, 7, 8, 9, 10], where the cost of a PIR scheme is measured by the total amount of communication between the user and the databases, i.e., the sum of lengths of each query string (upload) and each answering string (download). However, for the traditional Shannon theoretic formulation, where message size is allowed to be arbitrarily large, the upload cost is negligible compared to the download cost [11]². In this work we adopt the Shannon theoretic formulation, so that we focus on the download cost, measured relative to the message size. For the example presented above, each message is 1 bit and we download a total of 2 bits (one from each database), so that the download cost is 2 bits per message bit. The reciprocal of download cost is the rate, i.e., the number of bits of desired information that is privately retrieved per downloaded information bit. The maximum rate possible for the PIR problem is its information theoretic capacity C . For the example presented earlier, the private information retrieval rate is $\frac{1}{2}$, meaning that 1 bit of desired information is retrieved from every 2 downloaded bits. In general, for arbitrary N and K , the best previously known achievable rate for PIR, reported in [12], is $1 - \frac{1}{N}$. Since 1 is a trivial outer bound on capacity, we know that $1 \geq C \geq 1 - \frac{1}{N}$. The bounds present a reasonable approximation of capacity for large number of databases. However, in this work, we seek the *exact* information theoretic capacity C of the PIR problem, for *arbitrary* number of messages K and *arbitrary* number of databases N .

The problem statement is presented in Section 2. The exact capacity of PIR is characterized in Section 3. Section 4 presents a novel PIR scheme, and Section 5 provides the information theoretic outer bound to establish its optimality. Section 6 contains a discussion of the results and we conclude in Section 7.

Notation: For a positive integer Z , we use the notation $[Z] = \{1, 2, \dots, Z\}$.

2 Problem Statement

There are K messages W_1, \dots, W_K . The messages are independent and of the same size, i.e.,

$$H(W_1, \dots, W_K) = H(W_1) + \dots + H(W_K), \quad (3)$$

$$H(W_1) = \dots = H(W_K). \quad (4)$$

²The justification argument (traces back to Proposition 4.1.1 of [2]) is that the upload cost does not scale with the message size. This is because we can reuse the original query functions for each part of the message.

There are N databases, and each database stores all the messages W_1, \dots, W_K . A user wants to retrieve $W_i, i \in [K]$ privately, i.e., without revealing anything about the message identity i to any of the databases.

To retrieve W_i privately, the user first generates N queries $Q_1^{[i]}, \dots, Q_N^{[i]}$, where the superscript denotes the desired message index. Each query $Q_n^{[i]}, n \in [N]$ is a random variable with finite support. The queries are independent of the messages,

$$I(W_1, \dots, W_K; Q_1^{[i]}, \dots, Q_N^{[i]}) = 0. \quad (5)$$

The user sends query $Q_n^{[i]}$ to the n -th database. After receiving $Q_n^{[i]}$, the n -th database generates an answering string $A_n^{[i]}$, which is a deterministic function of $Q_n^{[i]}$ and the data stored (i.e., all messages W_1, \dots, W_K),

$$H(A_n^{[i]} | Q_n^{[i]}, W_1, \dots, W_K) = 0. \quad (6)$$

Each database returns to the user its answer $A_n^{[i]}$. From all answers $A_1^{[i]}, \dots, A_N^{[i]}$, the user can decode the desired message W_i ,

$$[\text{Correctness}] \quad H(W_i | A_1^{[i]}, \dots, A_N^{[i]}, Q_1^{[i]}, \dots, Q_N^{[i]}) = 0. \quad (7)$$

To satisfy the privacy constraint that each database learns nothing about the desired message index i information theoretically, each query $Q_n^{[i]}$ must be independent of i ,

$$[\text{Privacy}] \quad I(Q_n^{[i]}; i) = 0, \forall n \in [N]. \quad (8)$$

As the answering string is a deterministic function of the query and all messages, the answering string must be independent of i as well,

$$I(A_n^{[i]}; i) = 0, \forall n \in [N]. \quad (9)$$

The metric that we study in this paper is the PIR rate R , which characterizes how much information can be privately retrieved per download channel use. It is defined as the ratio between the size of the desired message and the total download cost,

$$R \triangleq \frac{H(W_i)}{\sum_{n=1}^N H(A_n^{[i]})}. \quad (10)$$

The PIR rate is the reciprocal of download efficiency. We aim to characterize the optimal (maximum) PIR rate, i.e., PIR capacity, C , over all private information retrieval schemes.

3 Main Result: Capacity of Private Information Retrieval

The following theorem states the main result.

Theorem 1 *For the private information retrieval problem with K messages and N databases, the private information retrieval capacity is*

$$C = (1 + 1/N + 1/N^2 + \dots + 1/N^{K-1})^{-1}. \quad (11)$$

The capacity is always strictly higher than the previously best known achievable rate of $1 - 1/N$. However, the capacity is a strictly decreasing function of the number of messages, K , and when the number of messages approaches infinity, the capacity approaches $1 - 1/N$. On the other hand, the capacity is strictly increasing in the number of databases, N . As the number of databases approaches infinity, the capacity approaches 1.

4 Proof of Theorem 1: Achievability

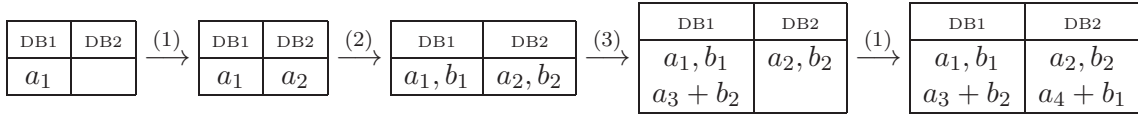
The capacity achieving PIR scheme has a myopic or greedy character, in that it starts with a narrow focus on the retrieval of the desired message bits from the first database, but grows into a full fledged scheme based on iterative application of three ideas:

- (1) *Enforcing Symmetry Across Databases*
- (2) *Enforcing Message Symmetry within the Query to Each Database*
- (3) *Exploiting Side Information of Undesired Messages to Retrieve New Desired Information*

To illustrate how these ideas work together in an iterative fashion, let us begin with a few simple examples corresponding to small values of K and N , and then generalize it to arbitrary K and N .

4.1 $K = 2$ messages, $N = 2$ Databases

Let $[a_k]$ represent bits from the desired message, and $[b_k]$ the bits from the undesired message. We start with a query that requests the first bit a_1 from the first database (DB1). Applying database symmetry, we simultaneously request a_2 from the second database (DB2). Next, we enforce message symmetry, by including queries for b_1 and b_2 as the counterparts for a_1 and a_2 . Now we have side information of b_2 from DB2 to be exploited in an additional query to DB1, which requests a new desired information bit a_3 mixed with b_2 . Finally, applying database symmetry we have the corresponding query $a_4 + b_1$ for DB2. At this point the queries satisfy symmetry across databases, message symmetry within the query to each database, and all undesired side information is exploited, so the construction is complete.



Privacy is ensured by mapping (a_1, a_2, a_3, a_4) to a random permutation of desired message bits, while (b_1, b_2, b_3, b_4) is mapped to a random permutation of the undesired message bits. Specifically, consider 4 bits from each message, so that $W_1 = (w_1(1), w_1(2), w_1(3), w_1(4))$, and $W_2 = (w_2(1), w_2(2), w_2(3), w_2(4))$. Let π_1 and π_2 be two i.i.d. uniform permutations of the indices 1, 2, 3, 4, generated privately by the user. To determine the actual query based on this construction, the user sets, $\forall k \in \{1, 2, 3, 4\}$,

$$a_k = w_1(\pi_1(k)), \quad b_k = w_2(\pi_2(k)), \quad \text{if } W_1 \text{ is the desired message,} \quad (12)$$

$$a_k = w_2(\pi_2(k)), \quad b_k = w_1(\pi_1(k)), \quad \text{if } W_2 \text{ is the desired message.} \quad (13)$$

Thus, regardless of the desired message, each database is asked for one randomly chosen bit of each message and a sum of a different pair of randomly chosen bits from each message. However, there remains a subtle issue related to the ordering of the queries.³ Note that if the queries are uploaded to each database exactly as presented, then the database will immediately be able to identify the first requested bit as belonging to the desired message. Because of the symmetry of messages, this is easily remedied by randomizing the order of queries within each database. For example, if

³There are other, more efficient ways of randomizing the queries, if the upload cost is of concern. Please see the discussion in Section 6.

the user chooses one of the two formats below, each with probability $1/2$, by privately flipping a fair coin, then each requested bit in the query to a database is equally likely to be the desired or undesired message bit.

DB1	DB2	DB1	DB2
a_1, b_1	a_2, b_2	b_1, a_1	b_2, a_2
$a_3 + b_2$	$a_4 + b_1$	$b_2 + a_3$	$b_1 + a_4$

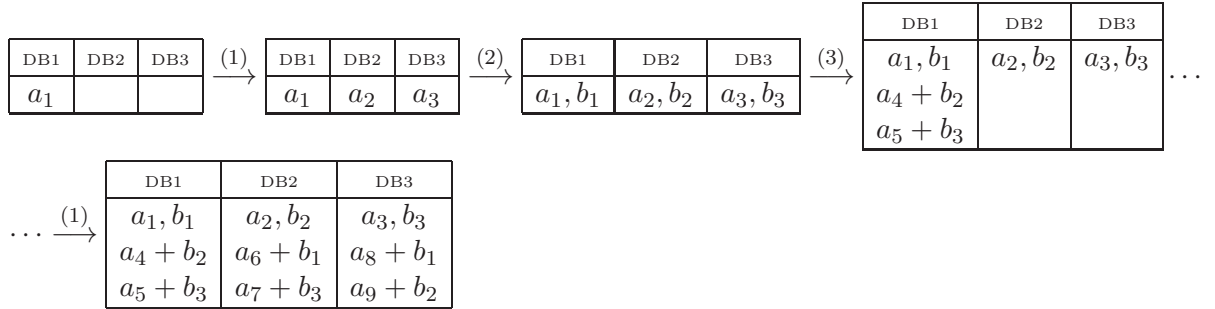
At this point, since all possible queries are equally likely regardless of desired message index, privacy is guaranteed.

To verify correctness, note that every desired bit is either downloaded directly or added with known side information which can be subtracted to retrieve the desired bit value. Thus, the desired message bits are successfully recoverable from the downloaded information.

Finally, consider the rate of this scheme. The total number of downloaded bits is 6, corresponding to $a_1, a_2, b_1, b_2, a_3 + b_2, a_4 + b_1$. The number of desired bits is 4, corresponding to a_1, a_2, a_3, a_4 . Thus, the rate of this scheme is $4/6 = 2/3$ which matches the capacity for this case.

4.2 $K = 2$ Messages, $N = 3$ Databases

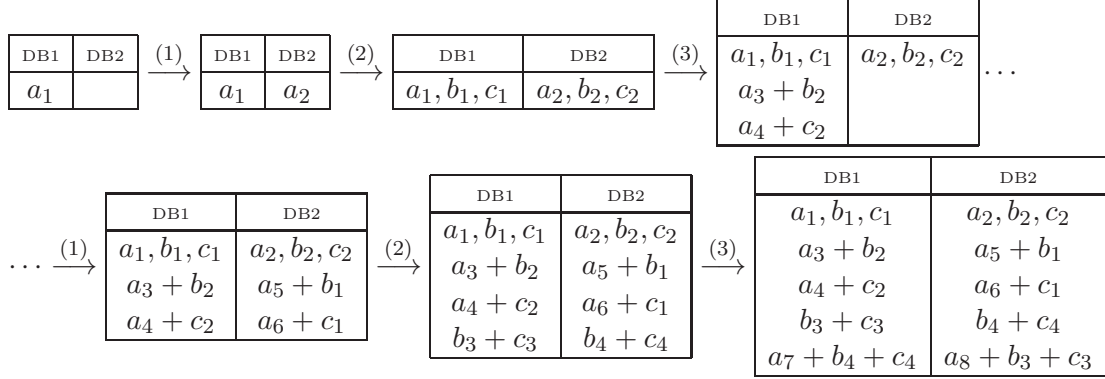
The construction of the optimal PIR scheme for $K = 2, N = 3$ is illustrated below, following the same iterative procedure as before.



Note that in the side information exploitation step, every downloaded bit from all other databases that is a sum of only undesired bits, is utilized by mixing with a new desired message bit to produce an additional query for DB1. Privacy is ensured by mapping (a_1, a_2, \dots, a_9) to a random permutation of the desired message bits, while (b_1, b_2, \dots, b_9) are mapped to a random permutation of the undesired message bits. The order of the queries within each database is also randomized. With these randomized mappings, since each possible query is equally likely regardless of desired message, privacy is ensured. Correctness is straightforward since each desired data bit is only added with known side information which can be subtracted to retrieve the desired data bit. Finally, this construction retrieves 9 desired message bits out of a total of 12 downloaded bits, so its rate is $9/12 = 3/4$, which matches the capacity for this case.

4.3 $K = 3$ Messages, $N = 2$ Databases

The construction of the optimal PIR scheme for $K = 2, N = 3$ is illustrated below.



The a_k bits are mapped to a random permutation of desired message bits, b_k and c_k are mapped to random permutations of the two remaining undesired messages and the order of queries is randomized as well. The rate achieved by this scheme matches the capacity for this case, $4/7$.

Remark: Note that if we reduce the number of messages, e.g., by setting all $c_k = 0$, then the scheme becomes the same as the optimal scheme for $K = 2$ messages that was presented earlier in Section 4.1. Indeed, this is a remarkable property of our optimal PIR scheme. It remains optimal over any subset of messages as well.

4.4 Arbitrary number of messages K , Arbitrary number of databases N

For arbitrary K, N , we follow the same iterative procedure. We start by downloading one desired bit a_1 from DB1. Enforcing symmetry across databases we download a new a_k from each database. Enforcing symmetry of messages, we also download one bit of each undesired message from each database. The next step is to utilize all the non-desired bits downloaded from all other databases, by mixing them with a new desired bit to produce additional queries for DB1. Note that there are $(N-1)(K-1)$ non-desired bits obtained from DB 2 to DB N , so we create $(N-1)(K-1)$ additional queries for DB1, each of which retrieves a new desired bit. Then we invoke symmetry of databases to include corresponding queries for each database. Next we invoke the symmetry of the messages by including queries comprised of sums of new bits of every pair of undesired messages. Since within each database we downloaded $(N-1)(K-1)$ equations, all of which contain new desired bits, we introduce the same number of equations $((N-1)(K-1))$ for every undesired message. Therefore, at this step, we download $(N-1)\binom{K}{2}$ additional equations from each database, each of which is a sum of two new undesired bits. We continue the iterative process by once again invoking database symmetry to determine the downloads from DB 2 to DB N , mirroring that of DB1. Then we utilize each of these new downloaded combinations of undesired message bits from other databases, by mixing each with a new desired bit to produce a new query for DB1. We then invoke another round of symmetry of databases, symmetry of messages and utilize side information. We repeat the process until we arrive at the situation where the queries are symmetric across both messages and databases, the download equations are sums of bits of all messages, and there are no unutilized combinations of only undesired bits. This happens because in the last step the equations are comprised of combinations of bits from all messages. We summarize the construction in the following algorithm.

- Step 1: Initialization. Download one desired bit a_1 from DB1.
- Step 2: Invoke symmetry across databases to determine corresponding downloads from DB 2 to DB N .
- Step 3: Invoke symmetry of messages to determine additional downloaded equations (comprised only of undesired bits) from each database.
- Step 4: Mix each new downloaded equation added in the previous step from DB 2 to DB N , with one new desired bit to produce additional downloads from DB1.
- Step 5: Go back to Step 2 and run Step 2 to Step 4 a total of $(K - 1)$ times.

Once the construction is complete, the randomized mapping of message bits and the randomization of the order of queries guarantees privacy. Correctness is guaranteed because, as usual, each desired bit is only mixed with sums of undesired bits that are directly downloaded from other databases, so they can be removed to retrieve the desired bits.

From each database, and for each $k \in \{1, 2, \dots, K\}$, this algorithm downloads $(N - 1)^{k-1} \binom{K}{k}$ equations that are comprised of sums of k bits, out of which $(N - 1)^{k-1} \binom{K-1}{k-1}$ involve desired data bits. Therefore the ratio of the number of total downloaded bits to the number of desired bits is

$$\frac{1}{R} = \frac{\binom{K}{1} + (N - 1)\binom{K}{2} + (N - 1)^2\binom{K}{3} + \dots + (N - 1)^{K-1}\binom{K}{K}}{\binom{K-1}{0} + (N - 1)\binom{K-1}{1} + (N - 1)^2\binom{K-1}{2} + \dots + (N - 1)^{K-1}\binom{K-1}{K-1}} \quad (14)$$

$$= 1 + \frac{\binom{K-1}{1} + (N - 1)\binom{K-1}{2} + (N - 1)^2\binom{K-1}{3} + \dots + (N - 1)^{K-2}\binom{K-1}{K-1}}{\binom{K-1}{0} + (N - 1)\binom{K-1}{1} + (N - 1)^2\binom{K-1}{2} + \dots + (N - 1)^{K-1}\binom{K-1}{K-1}} \quad (15)$$

$$= 1 + \frac{\frac{1}{N-1} \left[(N - 1)\binom{K-1}{1} + (N - 1)^2\binom{K-1}{2} + (N - 1)^3\binom{K-1}{3} + \dots + (N - 1)^{K-1}\binom{K-1}{K-1} \right]}{N^{K-1}} \quad (16)$$

$$= 1 + \frac{\frac{1}{N-1} (N^{K-1} - 1)}{N^{K-1}} = 1 + \frac{\frac{1}{N} (1 - \frac{1}{N^{K-1}})}{1 - \frac{1}{N}} \quad (17)$$

$$= 1 + \frac{1}{N} + \frac{1}{N^2} + \dots + \frac{1}{N^{K-1}} \quad (18)$$

Thus, the PIR rate achieved by the scheme always matches the capacity.

We conclude this section with another example, the PIR construction for the $K = 3, N = 3$ setting, which achieves the capacity $27/39 = 9/13$.

DB1	DB2	DB3
a_1	a_2	a_3
b_1	b_2	b_3
c_1	c_2	c_3
$a_4 + b_2$	$a_8 + b_1$	$a_{12} + b_1$
$a_5 + c_2$	$a_9 + c_1$	$a_{13} + c_1$
$a_6 + b_3$	$a_{10} + b_3$	$a_{14} + b_2$
$a_7 + c_3$	$a_{11} + c_3$	$a_{15} + c_2$
$b_4 + c_4$	$b_6 + c_6$	$b_8 + c_8$
$b_5 + c_5$	$b_7 + c_7$	$b_9 + c_9$
$a_{16} + b_6 + c_6$	$a_{20} + b_4 + c_4$	$a_{24} + b_4 + c_4$
$a_{17} + b_7 + c_7$	$a_{21} + b_5 + c_5$	$a_{25} + b_5 + c_5$
$a_{18} + b_8 + c_8$	$a_{22} + b_8 + c_8$	$a_{26} + b_6 + c_6$
$a_{19} + b_9 + c_9$	$a_{23} + b_9 + c_9$	$a_{27} + b_7 + c_7$

5 Proof of Theorem 1: Converse

5.1 Preliminaries

For compact notation, let us define

$$\mathcal{Q} \triangleq \{Q_n^{[k]} : k \in [K], n \in [N]\} \quad (19)$$

$$A_{n_1:n_2}^{[k]} \triangleq \{A_{n_1}^{[k]}, A_{n_1+1}^{[k]}, \dots, A_{n_2}^{[k]}\}, n_1 \leq n_2, n_1, n_2 \in [N], k \in [K] \quad (20)$$

Without loss of generality we make the following simplifications.

1. We assume that $\forall k \in [K], A_1^{[k]} = A_1, Q_1^{[k]} = Q_1$, i.e., the query and answering string for the first database is the same, regardless of which message is requested. There is no loss of generality in this assumption because the queries and answering strings for a given database are independent of the desired message index.
2. Given any PIR scheme, without reducing its rate, we can make it symmetric by replicating it for each permutation of databases and messages and combining the replicated schemes (essentially, time-sharing between all permutations of the scheme). So without loss of generality we will assume that the PIR scheme is symmetric in this sense. Subject to this assumption of symmetry, the rate of a PIR scheme is

$$R = \frac{H(W_1)}{NH(A_1)}. \quad (21)$$

Some consequences of the symmetry of the PIR scheme are formalized in the following lemma.

Lemma 1 *Without loss of generality, we have*

$$H(A_1^{[1]}|W_2, \dots, W_K, \mathcal{Q}) = \dots = H(A_N^{[1]}|W_2, \dots, W_K, \mathcal{Q}) \quad (22)$$

$$H(A_1|\mathcal{Q}) = H(A_n^{[k]}|\mathcal{Q}), \forall k \in [K], n \in [N] \quad (23)$$

$$H(A_1|W_2, \dots, W_K, \mathcal{Q}) \geq \frac{1}{N}H(W_1) \quad (24)$$

Proof: (22) follows directly from the symmetry across databases. Combined with the symmetry across messages, we have (23). For (24), note that from $A_{1:N}^{[1]}$, we can decode W_1 . From Fano's inequality, we have

$$\begin{aligned} H(W_1) &= I(A_{1:N}^{[1]}; W_1|W_2, \dots, W_K, \mathcal{Q}) = H(A_{1:N}^{[1]}|W_2, \dots, W_K, \mathcal{Q}) \\ &\leq \sum_{n=1}^N H(A_n^{[1]}|W_2, \dots, W_K, \mathcal{Q}) = NH(A_1|W_2, \dots, W_K, \mathcal{Q}) \end{aligned}$$

where the first line follows from the fact that the answers are deterministic functions of the messages and queries, and the second line is due to the property that dropping conditioning does not reduce entropy, and (22). Thus, (24) is proved. \blacksquare

The proof of outer bound for Theorem 1 is based on an induction argument. To set up the induction, we will prove the outer bound for $K = 1$ and for $K = 2$, each for arbitrary N , and then proceed to the case of arbitrary K .

5.2 $K = 1$ Message, N Databases

This case is straightforward, because if there is only one message, then the query is trivially independent of the message index. The user only needs to download his desired message bits from each database. As expected, the rate must be at least 1.

$$H(W_1) = H(W_1|\mathcal{Q}) \quad (25)$$

$$\leq I(A_1, A_2^{[1]}, \dots, A_N^{[1]}; W_1|\mathcal{Q}) \quad (26)$$

$$\leq H(A_1, A_2^{[1]}, \dots, A_N^{[1]}|\mathcal{Q}) \quad (27)$$

$$\leq \sum_{n=1}^N H(A_n^{[1]}|\mathcal{Q}) \quad (28)$$

$$= NH(A_1|\mathcal{Q}) \quad (29)$$

$$\Rightarrow R = \frac{H(W_1)}{NH(A_1)} \leq \frac{H(W_1)}{NH(A_1|\mathcal{Q})} = 1 \quad (30)$$

5.3 $K = 2$ Messages, N Databases

$$2H(W_1) = H(W_1, W_2|\mathcal{Q}) \quad (31)$$

$$\leq I(A_1, A_{2:N}^{[1]}, A_{2:N}^{[2]}; W_1, W_2|\mathcal{Q}) \quad (32)$$

$$= H(A_1, A_{2:N}^{[1]}, A_{2:N}^{[2]}|\mathcal{Q}) \quad (33)$$

$$= H(A_1, A_{2:N}^{[2]}|\mathcal{Q}) + H(A_{2:N}^{[1]}|A_1, A_{2:N}^{[2]}, \mathcal{Q}) \quad (34)$$

$$\leq NH(A_1|\mathcal{Q}) + H(A_{2:N}^{[1]}|A_1, A_{2:N}^{[2]}, \mathcal{Q}) \quad (35)$$

$$\leq NH(A_1|\mathcal{Q}) + H(A_{2:N}^{[1]}|A_1, A_{2:N}^{[2]}, W_2, \mathcal{Q}) \quad (36)$$

$$\leq NH(A_1|\mathcal{Q}) + H(A_{2:N}^{[1]}|A_1, W_2, \mathcal{Q}) \quad (37)$$

$$= NH(A_1|\mathcal{Q}) + H(A_{1:N}^{[1]}|W_2, \mathcal{Q}) - H(A_1|W_2, \mathcal{Q}) \quad (38)$$

$$\leq NH(A_1|\mathcal{Q}) + H(W_1) - H(W_1)/N \quad (39)$$

$$\Rightarrow H(W_1) \left(1 + \frac{1}{N}\right) \leq NH(A_1|\mathcal{Q}) \quad (40)$$

$$\Rightarrow R = \frac{H(W_1)}{NH(A_1)} \leq \frac{H(W_1)}{NH(A_1|\mathcal{Q})} \leq \left(1 + \frac{1}{N}\right)^{-1} \quad (41)$$

where (33) is due to the fact that the answering strings are deterministic functions of the messages and queries. (35) follows from the chain rule, the fact that conditioning reduces entropy, and (23). (36) is due to the fact that W_2 is a function of $A_1^{[1]}, A_{2:N}^{[2]}$. In (39) the second term follows from the fact that from $A_{1:N}^{[1]}$ we can decode W_1 , and the last term is due to (24). The outer bound proof for $K = 2$ messages setting is complete.

An intuitive understanding of the outer bound proof is as follows. Overall, we download $H(W_1)/R$ amount of information, out of which $H(W_1)$ is the desired message and the remaining $(1/R - 1)H(W_1)$ is interference. Suppose we request W_2 . Then, the interference is due to W_1 , and $(1/R - 1)H(W_1)$ is the amount of undesired information about W_1 contained in the answering strings. From (24), we know that this interference is at least $H(W_1)/N$, so that $(1/R - 1)H(W_1) \geq H(W_1)/N$, and therefore $1/R \geq 1 + 1/N$.

5.3.1 $K \geq 3$ Messages, N Databases

From $A_1, A_{2:N}^{[1]}, \dots, A_{2:N}^{[K]}$, we can decode all K messages W_1, \dots, W_K . From Fano's inequality, we have

$$\begin{aligned} & KH(W_1) \\ &= H(W_1, \dots, W_K|\mathcal{Q}) \end{aligned} \quad (42)$$

$$\leq I(A_1, A_{2:N}^{[1]}, \dots, A_{2:N}^{[K]}; W_1, \dots, W_K|\mathcal{Q}) \quad (43)$$

$$= H(A_1, A_{2:N}^{[1]}, \dots, A_{2:N}^{[K]}|\mathcal{Q}) \quad (44)$$

$$\leq H(A_1, A_{2:N}^{[1]}|\mathcal{Q}) + H(A_{2:N}^{[2]}, \dots, A_{2:N}^{[K]}|A_1, A_{2:N}^{[1]}, \mathcal{Q}) \quad (45)$$

$$\leq NH(A_1|\mathcal{Q}) + H(A_{2:N}^{[2]}, \dots, A_{2:N}^{[K]}|A_1, A_{2:N}^{[1]}, W_1, \mathcal{Q}) \quad (46)$$

$$\leq NH(A_1|\mathcal{Q}) + H(A_{2:N}^{[2]}|W_1, \mathcal{Q}) + H(A_{2:N}^{[3]}, \dots, A_{2:N}^{[K]}|A_1, A_{2:N}^{[1]}, A_{2:N}^{[2]}, W_1, \mathcal{Q}) \quad (47)$$

$$\leq NH(A_1|\mathcal{Q}) + \sum_{n=2}^N H(A_n^{[2]}|W_1, \mathcal{Q}) + H(A_{2:N}^{[3]}, \dots, A_{2:N}^{[K]}|A_1, A_{2:N}^{[1]}, A_{2:N}^{[2]}, W_1, W_2, \mathcal{Q}) \quad (48)$$

$$\leq NH(A_1|\mathcal{Q}) + \sum_{n=2}^N H(A_n^{[2]}, A_{1:n-1}^{[1]}, A_{n+1:N}^{[1]}|W_1, \mathcal{Q}) + H(A_{2:N}^{[3]}, \dots, A_{2:N}^{[K]}|A_1, W_1, W_2, \mathcal{Q}) \quad (49)$$

$$\begin{aligned} &= NH(A_1|\mathcal{Q}) + \sum_{n=2}^N \left(H(A_n^{[2]}, A_{1:n-1}^{[1]}, A_{n+1:N}^{[1]}, W_1|\mathcal{Q}) - H(W_1|\mathcal{Q}) \right) \\ &\quad + H(A_1, A_{2:N}^{[3]}, \dots, A_{2:N}^{[K]}|W_1, W_2, \mathcal{Q}) - H(A_1|W_1, W_2, \mathcal{Q}) \end{aligned} \quad (50)$$

$$\begin{aligned}
&\leq NH(A_1|\mathcal{Q}) + \sum_{n=2}^N \left(H(A_n^{[2]}, A_{1:n-1}^{[1]}, A_{n+1:N}^{[1]}|\mathcal{Q}) + \underbrace{H(W_1|A_n^{[2]}, A_{1:n-1}^{[1]}, A_{n+1:N}^{[1]}, \mathcal{Q})}_{=0} - H(W_1) \right) \\
&\quad + H(W_3, \dots, W_K) - H(A_1|W_1, W_2, \mathcal{Q}) \tag{51} \\
&\leq NH(A_1|\mathcal{Q}) + (N-1)(NH(A_1|\mathcal{Q}) - H(W_1)) + (K-2)H(W_1) - H(A_1|W_1, W_2, \mathcal{Q}) \tag{52} \\
&\leq N^2H(A_1|\mathcal{Q}) + (K-N-1)H(W_1) - H(A_1|W_1, W_2, \mathcal{Q}) \tag{53} \\
&\Rightarrow NH(A_1|\mathcal{Q}) \geq H(W_1) \left(1 + \frac{1}{N} \right) + \frac{1}{N}H(A_1|W_1, W_2, \mathcal{Q}) \tag{54}
\end{aligned}$$

where (46) is due to the fact that W_1 is a deterministic function of $A_1, A_{2:N}^{[1]}$. (48) follows from the fact that W_2 is a deterministic function of $A_1, A_{2:N}^{[2]}$. In (51), the third term equals 0, because from $A_n^{[2]}, A_{1:n-1}^{[1]}, A_{n+1:N}^{[1]}$, we can decode W_1 , the second last term is due to the fact that from $A_1^{[1]}, A_{2:N}^{[3]}, \dots, A_{2:N}^{[K]}$, we can decode W_3, \dots, W_K . In (52), the second term follows from (23). To proceed, we note that for the last term of (54), conditioning on W_1, W_2 , the setting reduces to a PIR problem with $K-2$ messages and N databases. Thus, (54) sets up an induction argument, which claims that for the K messages setting,

$$NH(A_1|\mathcal{Q}) \geq H(W_1) \left(1 + \frac{1}{N} + \dots + \frac{1}{N^{K-1}} \right) \tag{55}$$

We have proved the basis cases of $K=1$ and $K=2$ in (29) and (40). Suppose now the bound (55) holds for $K-2$. Then plugging in (54), we have that the bound (55) holds for K . Since both the basis and the inductive step have been performed, by mathematical induction, we have proved that (55) holds for all K . The desired outer bound follows as

$$R = \frac{H(W_1)}{NH(A_1)} \leq \frac{H(W_1)}{NH(A_1|\mathcal{Q})} \leq \left(1 + \frac{1}{N} + \dots + \frac{1}{N^{K-1}} \right)^{-1} \tag{56}$$

Thus, the outer bound proof is complete.

6 Discussion

In this section we reflect upon some practical concerns beyond capacity.

Upload Cost

To ensure privacy, we appealed to randomization arguments. To specify the randomly chosen query to the databases incurs an upload cost. For large messages the upload cost is negligible relative to the download cost, so it was ignored in this work. However, if the upload cost is a concern then it could be optimized as well. Random permutations of message bits, and the randomized ordering of queries, are sufficient for privacy, but it is easy to see that the upload cost can be reduced by reducing the number of possibilities to be considered. For example, consider the $K=2$ messages, $N=2$ databases setting. We can group the bits, i.e., we can divide the 4 bits of each message into 2 groups, so that when we choose 2 bits, we only choose 2 bits from the same group. This reduces the choice to 1 out of 2 groups (rather than 2 out of 4 bits). Further, it may be possible to avoid random permutations among the chosen bits (group). For the same $K=2$ messages and $N=2$

databases example, we can fix the order within each group and the scheme becomes the following. We denote the messages bits as $W_1 = \{u_1, u_2, u_3, u_4\}$, $W_2 = \{v_1, v_2, v_3, v_4\}$.

	Prob. 1/2		Prob. 1/2	
	Want W_1	Want W_2	Want W_1	Want W_2
Database 1	$u_1, v_1, u_2 + v_2$	$u_1, v_1, u_2 + v_2$	$u_3, v_3, u_4 + v_4$	$u_3, v_3, u_4 + v_4$
Database 2	$u_4, v_2, u_3 + v_1$	$u_2, v_4, u_1 + v_3$	$u_2, v_4, u_1 + v_3$	$u_4, v_2, u_3 + v_1$

Note that regardless of which message is desired, the user is equally likely to request either $u_1, v_1, u_2 + v_2$ or $u_3, v_3, u_4 + v_4$ from DB1, and either $u_2, v_4, u_1 + v_3$ or $u_4, v_2, u_3 + v_1$ from DB2, so the scheme is private. However, each query is now limited to only 2 possibilities, thereby significantly reducing the upload cost. Also note that instead of storing all 8 bits that constitute the two messages, each database only needs to store 6 bits in this case, corresponding to the two possible queries that it may face. Reducing the *storage overhead* is an interesting question that has been explored by Fazeli, Vardy and Yaakobi in [13].

Another interesting question in this context is to determine the *upload constrained capacity*. An information theoretic perspective is still useful. For example, since we are able to reduce the upload cost for $K = 2, N = 2$ to two possibilities, one might wonder if it is possible to reduce the upload cost of the $K = 3, N = 2$ setting to 3 possibilities without loss of capacity. Let us label the three possible downloads from DB1 as f_1, f_2, f_3 and the three possible downloads from DB2 as g_1, g_2, g_3 . We wish to find out if the original PIR capacity of $4/7$ is still achievable under these upload constraints. As we show next, the capacity is strictly reduced. With uploads limited to choosing one out of only 3 possibilities, the upload constrained capacity of the $K = 3, N = 2$ setting is $1/2$ instead of $4/7$. Eliminating trivial degenerate cases, in this case there is no loss of generality in assuming that we can recover W_1 from any one of these three possibilities: $(f_1, g_1), (f_2, g_2), (f_3, g_3)$; we can recover W_2 from any one of these three possibilities: $(f_1, g_2), (f_2, g_3), (f_3, g_1)$; and we can recover W_3 from any one of these three possibilities: $(f_1, g_3), (f_2, g_1), (f_3, g_2)$. Then, for the optimal scheme we have

$$H(W_1) = I(W_1; f_1, g_1) \quad (57)$$

$$\leq 2H(A) - H(f_1, g_1|W_1) \quad (58)$$

$$\text{Similarly, } H(W_1) \leq 2H(A) - H(f_2, g_2|W_1) \quad (59)$$

$$\text{Adding the two, } 2H(W_1) \leq 4H(A) - H(f_1, g_1, f_2, g_2|W_1) \quad (60)$$

$$\leq 4H(A) - H(W_1, W_2, W_3|W_1) \quad (61)$$

$$\leq 4H(A) - H(W_2, W_3) \quad (62)$$

$$\Rightarrow C = H(W_1)/2H(A) \leq 1/2 \quad (63)$$

Here, $2H(A)$ is the total download. (61) follows because from f_1, g_1, f_2, g_2 we can recover all three messages. Thus, if the upload can only resolve one out of three possibilities for the query to each database, then the capacity of such a PIR scheme cannot be more than $1/2$, which is strictly smaller than the PIR capacity without upload constraints, $4/7$. In fact, the upload constrained capacity in this case is exactly $1/2$, as shown by the following achievable scheme which is interesting in its own right for how it fully exploits interference alignment. Suppose W_1, W_2, W_3 are symbols from a sufficiently large finite field (e.g., \mathbb{F}_5). Then the following construction works.

$$f_1 = W_1 + 2W_2 + W_3 \quad (64)$$

$$f_2 = W_1 + 4W_2 + 3W_3 \quad (65)$$

$$f_3 = 3W_1 + 4W_2 + 6W_3 \quad (66)$$

$$g_1 = W_1 + 4W_2 + 2W_3 \quad (67)$$

$$g_2 = 3W_1 + 4W_2 + 3W_3 \quad (68)$$

$$g_3 = 2W_1 + 4W_2 + 6W_3 \quad (69)$$

It is easy to verify that W_1 can be recovered from any one of $(f_1, g_1), (f_2, g_2), (f_3, g_3)$; W_2 can be recovered from any one of $(f_1, g_2), (f_2, g_3), (f_3, g_1)$; and W_3 can be recovered from any one of $(f_1, g_3), (f_2, g_1), (f_3, g_2)$. The reason we can recover the desired message symbol from two equations, even though all three message symbols are involved in those two equations, is because of this special construction, which forces the undesired symbols to align into one dimension in every case. Thus, the upload constrained capacity for $K = 3, N = 2$ when the randomness is limited to choosing one out of 3 possibilities, is $1/2$. Answering this question for arbitrary K, N and arbitrary upload constraints is an interesting direction for future work.

Message Size

The information theoretic formulation of the PIR problem allows the sizes of messages to grow arbitrarily large. A natural question is this – how large do we need each message to be for the optimal scheme. It is easy to see that in our scheme, each message consists of N^K bits. However, even for our capacity achieving PIR scheme, the size of a message may be reduced, e.g., if the assumption of symmetry is removed. As an example, for the same $K = 2$ messages and $N = 2$ databases setting, the following PIR scheme works just as well (still achieves the same capacity)⁴ when each message is only made up of 2 bits: $W_1 = (u_1, u_2)$, $W_2 = (v_1, v_2)$. However, here the download is not symmetric. 2 bits are downloaded from Database 1 and only 1 bit from Database 2.

	Prob. 1/2		Prob. 1/2	
	Want W_1	Want W_2	Want W_1	Want W_2
Database 1	u_1, v_2	u_1, v_2	u_2, v_1	u_2, v_1
Database 2	$u_2 + v_2$	$u_1 + v_1$	$u_1 + v_1$	$u_2 + v_2$

Determining the smallest message size needed to achieve the PIR capacity, or the message size constrained PIR capacity, is another interesting direction for future work.

7 Conclusion

Information theorists commonly study the optimal coding rates of communication problems dealing with a few messages, each carrying an asymptotically large number of bits, while computer scientists often study the computational complexity of problems dealing with an asymptotically large number of messages, each carrying only a few bits (e.g., 1 bit per message). The occasional crossover of problems between the two fields opens up exciting opportunities for new insights. A prominent example is the index coding problem [14], originally posed by computer scientists and recently studied from an information theoretic perspective. The information theoretic capacity

⁴Note that allowing asymmetry does not change the capacity of PIR.

characterization for the index coding problem is now recognized as perhaps one of the most important open problems in network information theory, because of its fundamental connections to a broad range of questions that includes topological interference management, network coding, distributed storage, hat guessing, and non-Shannon information inequalities. Like index coding, the PIR problem also involves non-trivial interference alignment principles and is related to problems like blind interference alignment [15] that have previously been studied in the context of wireless networks. In fact, it was the pursuit of these connections that brought us to the PIR problem [16]. Further, PIR belongs to another rich class of problems studied in computer science, with deep connections to oblivious transfer [17], instance hiding [18, 19, 20], and distributed computation with untrusted servers [21]. Bringing this class of problems into the domain of information theoretic studies holds much promise for new insights and fundamental progress. The characterization of the information theoretic capacity of Private Information Retrieval is a step in this direction.

References

- [1] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, “Private information retrieval,” in *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, 1995, pp. 41–50.
- [2] B. Chor, E. Kushilevitz, O. Goldreich, and M. Sudan, “Private Information Retrieval,” *Journal of the ACM (JACM)*, vol. 45, no. 6, pp. 965–981, 1998.
- [3] W. Gasarch, “A Survey on Private Information Retrieval,” in *Bulletin of the EATCS*. Citeseer, 2004.
- [4] S. Yekhanin, “Private Information Retrieval,” *Communications of the ACM*, vol. 53, no. 4, pp. 68–73, 2010.
- [5] R. Ostrovsky and W. E. Skeith III, “A Survey of Single-database Private Information Retrieval: Techniques and Applications,” in *Public Key Cryptography–PKC 2007*. Springer, 2007, pp. 393–411.
- [6] A. Ambainis, “Upper bound on the communication complexity of private information retrieval,” in *Automata, Languages and Programming*. Springer, 1997, pp. 401–407.
- [7] A. Beimel, Y. Ishai, and E. Kushilevitz, “General constructions for information-theoretic private information retrieval,” *Journal of Computer and System Sciences*, vol. 71, no. 2, pp. 213–247, 2005.
- [8] A. Beimel, Y. Ishai, E. Kushilevitz, and J.-F. Raymond, “Breaking the $\mathcal{O}(n^{1/(2k-1)})$ barrier for information-theoretic private information retrieval,” in *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science*. IEEE, 2002, pp. 261–270.
- [9] S. Yekhanin, “Locally Decodable Codes and Private Information Retrieval Schemes,” Ph.D. dissertation, Massachusetts Institute of Technology, 2007.
- [10] Z. Dvir and S. Gopi, “2-Server PIR with Sub-polynomial Communication,” *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC’15*, pp. 577–584, 2015.

- [11] T. H. Chan, S.-W. Ho, and H. Yamamoto, "Private Information Retrieval for Coded Storage," *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, pp. 2842–2846, 2015.
- [12] N. Shah, K. Rashmi, and K. Ramchandran, "One Extra Bit of Download Ensures Perfectly Private Information Retrieval," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, 2014, pp. 856–860.
- [13] A. Fazeli, A. Vardy, and E. Yaakobi, "Codes for distributed PIR with low storage overhead," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, 2015, pp. 2852–2856.
- [14] Z. Bar-Yossef, Y. Birk, T. S. Jayram, and T. Kol, "Index coding with side information," in *47th Annual IEEE Symposium on Foundations of Computer Science, 2006. FOCS '06.*, 2006, pp. 197 – 206.
- [15] S. A. Jafar, "Blind Interference Alignment," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 3, pp. 216–227, June 2012.
- [16] H. Sun and S. A. Jafar, "Blind Interference Alignment for Private Information Retrieval," *arXiv preprint arXiv:1601.07885*, 2016.
- [17] Y. Gertner, Y. Ishai, E. Kushilevitz, and T. Malkin, "Protecting data privacy in private information retrieval schemes," in *Proceedings of the thirtieth annual ACM symposium on Theory of computing.* ACM, 1998, pp. 151–160.
- [18] J. Feigenbaum, "Encrypting problem instances," in *Advances in Cryptology – CRYPTO85 Proceedings.* Springer, 1985, pp. 477–488.
- [19] M. Abadi, J. Feigenbaum, and J. Kilian, "On hiding information from an oracle," in *Proceedings of the nineteenth annual ACM symposium on Theory of computing.* ACM, 1987, pp. 195–203.
- [20] D. Beaver and J. Feigenbaum, "Hiding instances in multioracle queries," in *STACS 90.* Springer, 1990, pp. 37–48.
- [21] D. Beaver, J. Feigenbaum, J. Kilian, and P. Rogaway, "Locally random reductions: Improvements and applications," *Journal of Cryptology*, vol. 10, no. 1, pp. 17–36, 1997.